

Quarch Technology Ltd

Torridon Array Controller

Technical Manual

For use with:

QTL1079 - 28 Port Torridon Array Controller

Monday, 11 February 2019



Change History

1.0	2nd November 2010	Initial Release
1.2	20 November 2010	Added SD card support for stored Ethernet settings Added module list commands Improved self test functionality Added new style help screens
1.3	26 November 2010	Updated with new format common sections
1.4	29 April 2011	Updated with new firmware features
1.5	24 March 2014	Updated with ReST communications option
1.6	8 th December 2014	Brings this product in-line with 4-port controller
1.7	20 th February 2018	Added user access control information
1.8	11 th February 2018	Added quarchpy links to ReST section

Contents

Quarch Technology Ltd.....	1
Torridon Array Controller	1
Technical Manual.....	1
Change History	2
Contents.....	3
Introduction	5
Technical Specifications	6
Power Requirements:.....	6
Mechanical Characteristics:.....	7
Electrical Connections	8
Front Panel	8
Rear Panel.....	8
Control Interfaces	9
Serial Scripting	9
Serial Settings	9
USB.....	9
Ethernet (Telnet)	9
Connection	10
Ethernet (ReST).....	11
User Access Control (ReST Only)	12
Access control basics	12
Chaining Array Controllers.....	14
Array Controller Commands	15
Addressing Commands	16
Address Mapping.....	17

Saving and Loading Settings	19
Basic Commands.....	19
General Setup Commands.....	21
Addressing Commands	22
Ethernet Commands.....	24
Voltage Measurement Commands.....	25
User Access Control Commands.....	25
Voltage Measurements	27
Factory Default Startup State	28
Control Register Map	29

Introduction

The **Quarch Technology *Torridon* Array Controller - QTL1079**, allows centralized control of up to 28 Torridon Control Modules. Supported modules include:

- Standard Drive Control Modules
- Lite Drive Control Modules
- HS Lite Drive Control Modules
- High Speed Drive Control Modules
- Cable Pull Modules
- SBB Canister Control Modules
- Programmable Power Modules
- Physical Layer Switches

The Array Controller provides a simple terminal interface to the user, single commands or entire scripts can be sent concurrently to any number of attached Control Modules.

The operation of the Array Controller is almost transparent to the user. A minimal change to the command structure allows individual Control Modules to be addressed directly. Existing control scripts can be quickly modified to work with the new system.

Other features of the system include:

- Cascadable, up to 4 Array controllers can be linked in series
- Simplified cabling requirements
- LED indicators of Control Module status
- Updatable and future proof
- Rack mountable

This manual is designed to be used in conjunction with the “Torridon System User Manual” and the relevant Technical Manuals for the modules you are operating.

Technical Specifications

Power Requirements:

A 12v DC supply is provided to power the Array Controller. The Array Controller supplies power to attached Control Modules.

The system power consumption can be estimated as the sum of the power consumption of each component in the system:

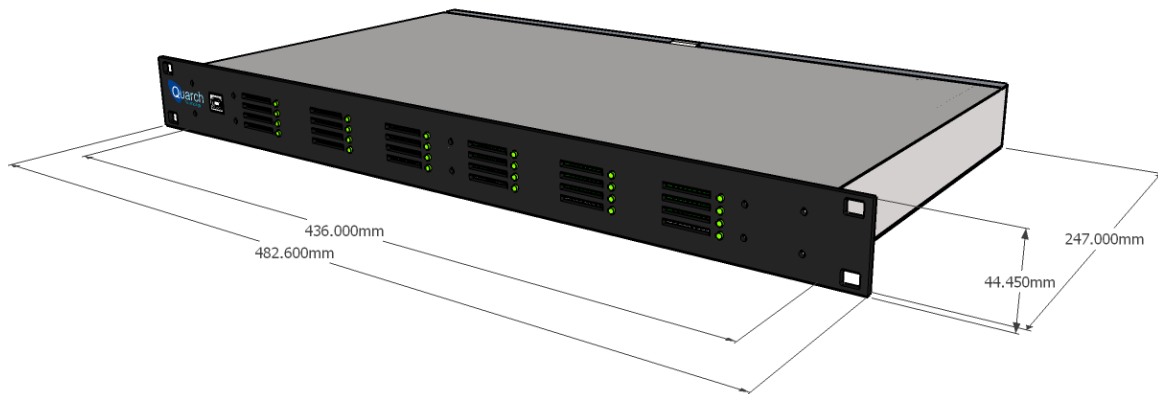
Control Module Type	Watts
Drive Control Module	1
Cable Pull Module	1
SBB Control Module	3

The system comes as default with a 60 Watt, 12v supply. This will be suitable for most combinations of current Quarch Control Modules.

The Array Controller is internally fused (auto resettable) and current limited. Only original Quarch manufactured or approved control modules should be used with the system.

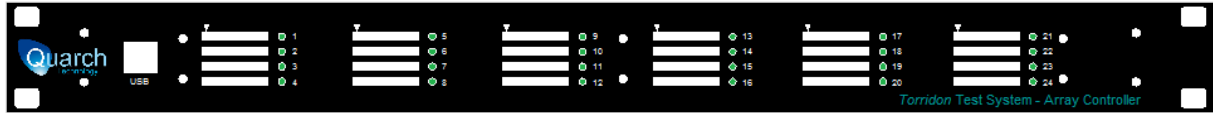
Mechanical Characteristics:

The Array Controller fits in a standard 19" rack and takes up 1U in height. The unit does not require any clear space above or below it for cooling.



Electrical Connections

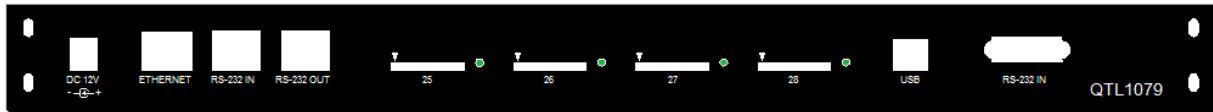
Front Panel



(left to right)

USB	USB-B, for USB connection to a PC (not recommended for normal operation due to the large number of USB devices present)* ¹
Ports 1-24	Control Module connections (Normally used for Drive Control Modules). A green LED indicates the module has been identified by the system and has good status. The LED blinks off as commands are received.

Rear Panel



(left to right)

DC 12v IN	2.1mm or 2.5mm Centre Positive Jack
Ethernet	RJ45, 10BaseT Ethernet connection
RS-232 IN	RJ45(RS232D), Serial control in. Connect to your terminal* ²
RS-232 OUT	RJ45, Chaining port. Connects to the next Array Controller
Ports 25-28	Control Module connections (Normally used for SBB or Power Supply Control Modules)
USB	USB-B, for USB connection to controlling PC (not recommended for normal operation due to the large number of USB devices present) * ¹
RS-232 IN	DB9, Serial control in. Connect to your terminal* ²

*¹ The front and rear USB ports are multiplexed, the rear port will be disabled when the front port is connected

*² These ports may not be connected at the same time

Control Interfaces

Serial Scripting

The main mode of control is via ASCII serial commands. These can be transmitted from an application such as HyperTerminal or generated from your own scripts or code.

Most of the commands you send will be directed at the Control Modules you are operating rather than the Array Controller itself. Please see the “Torridon System Manual” for details on how to operate the system.

Serial Settings

Setting	Required value
Baud Rate	19,200
Data Bits	8
Stop Bits	1
Handshaking	None or Hardware

USB

The USB sockets provide a connection to the Array Controller and also to every one of the 28 ports. This connection is only used for updating the firmware on the control modules. As some operating systems can be unstable with large numbers of USB devices connected, it is recommended to leave this unplugged until it is required.

Ethernet (Telnet)

Control via a Telnet terminal works in exactly the same way as the serial terminal described above.

Only one Telnet connection to the unit can be made at a time, further connections will be refused. An active Telnet connection will also lock out the serial terminal as only one connection can be active at a time.

If you want to prevent a Telnet connection from taking over control from the serial terminal, you can use the command: “**conf:terminal:lock on**”.

Connection

To use Telnet, connect the Array Controller to your network using an RJ-45 cable into the “ETHERNET” port on the rear of the unit.

By default, the Array Controller requires a DHCP server to obtain its IP address. If one is not available, the Ethernet connection will not work.

Once connected, you can obtain the IP address using the command ***“conf:ethernet:ip?”*** from the serial terminal.

The Array Controller also supports NETBIOS names. The name of the unit is based on its serial number. This can be found on the rear of the unit in the form: “QTL1079-xx-xxx”. The NETBIOS name will be:

QTL1079-xx-xxx

Where xx-xxx is the last digits of the serial number.

Connection to the Telnet terminal should be on Port 23 in the connection settings of your terminal program.

To set a static IP address, use the command ***“conf:ethernet:IP xxx.xxx.xxx.xxx”***. To make this change permanent between resets, see the section on “Saving and Loading Settings”

Ethernet (ReST)

Control via the ReST API allows you to communicate easily from any TCP/IP client. Physical connection should be made as given in the Telnet instructions above.

If you send a ReST command to the Array, it will lock out Serial/USB/Telnet communication during the execution and response of the command. ReST control is compatible with multiple users, unlike Telnet which only allows a single connection at a time.

ReST commands are almost identical to the standard terminal mode commands. For example, the command “*run:power up <1>*” becomes a URL: “*http://192.168.1.55/run:power up%20<1>*”. The response will be returned as a raw ASCII format data within a basic webpage.

NOTE: spaces in commands need to be escaped with the %20 character!

For a simple test, try the Firefox RESTclient add-in. This will allow you to send commands and see the responses.

The most common way to use ReST control is with the Quarch Python library ‘quarchpy’. This can be installed easily with the command:

Python -m pip install quarchpy

The following application note shows how to use the quarchpy module in a set of simple examples. All communication types are supported by quarchpy:

<https://quarch.com/file/an-006-python-control-quarch-modules>

User Access Control (ReST Only)

As ReST access permits multiple users to send commands at the same time, we provide the ability to control user access to the various ports over the ReST communication channel.

This feature was implemented as of Firmware version:

3.76 for -01, -02 and -03 modules

4.003 for -04 modules

You will need to upgrade your controller if you are running on an older version.

Access Control is designed to avoid user accidentally interrupting another users test. It does this by limiting a user's control to a specific ranges of modules. It is NOT intended as a security mechanism, and will not function as one.

Access control basics

By default, access control is disabled and the controller will accept any command, as usual. Even when enabled, USB, serial and Telnet control will respond as normal, as these are 'single user' access mechanisms. This means you can always recover control via USB or similar if you forget the user IDs.

To enable/disable access control:

CONfig:USER:CONtrol ON

CONfig:USER:CONtrol OFF

Up to 15 different users can be created, each with different access rights. Each user can have access over any selection of ports and can optionally be an 'Admin' which allows them to perform actions such as resetting the controller and

Each user is identified by a unique 'UserID' consisting of exactly 8 characters. When communicating over ReST with user control turned on, this ID is placed at the start of the command:

Command with access control turned OFF

Run:power up <1-4>

Command with access control turned ON

andy1234@Run:power up <1-4>

In every case, the format for commands is the 8 character UserID followed by '@', then the standard command. Failure to add the UserID, or supplying a UserID that does not exist, will cause an error.

To create a user, we create it in one of the available slots:

CONFig:USER:3:SET mike1002

We then grant permissions to the required ports (1-8 and 24 in this example):

CONFig:USER:3:GRANT 1 8

CONFig:USER:3:GRANT 24

The user 'mike1002' will now have control over modules on ports 1-8 and port 24. Trying to access any other port will result in an error and the command will not execute.

See the command listing later in this manual for the full access control command set.

Chaining Array Controllers

Up to 4 Array Controllers can be connected together to provide a single point of control for many modules.

To connect 2 Array Controllers, plug a standard RJ-45 patch cable (straight through) between the RS-232 OUT port of the first controller and the RS-232 IN port of the second. Repeat until all Array Controllers are linked. The green 'link' light should appear on each port to show that the connection has been made.

You can now send commands to the first Array Controller as normal. See the 'Address Mapping' section below for the list of default addresses for the ports. Note that that by default, port 1 of the second array will be address '30'.

The 'Address Mapping' section also describes how to renumber the addresses using 'soft mapping' if this is required.

Array Controller Commands

The main purpose of the Array Controller is to route commands to one or more Control Modules; this is achieved through an address suffix described in the Addressing Commands section below. The Array Controller itself accepts a number of commands which are described in the following section.

NOTE: only the commands for the Array Controller itself are shown here. For Control Module commands, you must refer to the manual for that Module.

The Array Controller has a 64 character line buffer; any command sent MUST not exceed this limit, which would result in a failed command. The only time you are likely to exceed this limit is if you have used a very long address list.

These commands are based on the SCPI style control system that is used by many manufacturers of test instruments. The entire SCPI specification has NOT been implemented but the command structure will be very familiar.

- SCPI commands are case-insensitive
- SCPI commands are in a hierarchy separated by ':' (LEVe1:LEVe2:LEVe3)
- Most words have a short form (e.g. 'register' shortens to 'reg'). This will be documented as REGister, where the short form is shown in capitals.
- Query commands that return a value all have a '?' on the end
Commands with a preceding '*' are basic control commands, found on all SCPI devices

Addressing Commands

Commands sent through the Array Controller must have an address list suffix or the Array Controller will try to execute the command rather than passing it along to an attached Control Module. To send a command to any Control Module in the system you must know its address.

An address is given in the form <n> where 'n' is the decimal port number

Array controller: Any command without an address list will be executed by the Array Controller

Control Module Ports: By default these have a numerical address, matching the number written next to the port (from <1> to <28> for the first array)

Commands may be sent to multiple Control Modules, either as a group <1,2,3,4> or as a range <1-4>. Groups and ranges may also be combined <1-4,8,16> in any order. Ranges **MUST** be low-to-high.

To send a command to a Control Module, simply append the address list to the end of the command:

“Reg:Read 0xFF <1,2,3,8-15>”

This will read the 0xFF register on the 8 addressed Control Modules (1,2,3,4,8,9,10,11,12,13,14 and 15). Each Control Module will respond with its address number and the result of the query:

“1.0:0x11”

“2.0:0x11”

“3.0:0x11”

“...etc...”

Note that the response address is in the form “a.b:” where ‘a’ is the port number and ‘b’ can be ignored (this is used internally and will always be 0 when talking to Control Modules)

The order of the returns is fixed so that address 1.0 will be the first port to respond.

Address Mapping

The default address map for up to 4 chained Array Controllers:

Hard Address	Device(s)	Default Soft Address
No address list or <0>	1 st Array Controller	0
1..28	Ports 1..28 on 1 st Array Controller	1..28
29	2 nd Array Controller	800
30..57	Ports 1..28 on 2 nd Array Controller	29..56
58	3 rd Array Controller	801
59..86	Ports 1..28 on 3 rd Array Controller	57..84
87	4 th Array Controller	802
88..115	Ports 1..28 on 4 th Array Controller	85..112

NOTE: The Array Controllers are designed to function without modification to the default address maps, the following commands are provided for advanced users.

It is possible to re-assign 'Soft Addresses' of ports on the Array Controller in order to simplify cabling or suit your own numbering requirements:

'Hard Addresses' refer to the fixed numbers on the front of the Array Controller. If two or more Array Controllers are chained, then the second will start numbering two above the last port on the first. This is due to the extra "Array Controller" address that is not used in normal operation.

'Soft Addresses' refer to user assigned addresses. These can be any integer number up to 999. More than one 'hard' addressed port can be set to the same 'soft' number although this is not normally recommended as it will make it difficult to understand the responses.

The user can write to the mapping table to set a new address arrangement. When satisfied, the 'Activate' command sends the new mappings to all ports.

The 'Reset' command sets the mapping table back to its default 1:1 setting **AND** sends the new mappings to all ports. This will revert the address mapping to its power on state.

The 'Flash' command is a useful debug command which will flash the light of a selected port to help you find where to connect your cable to!

See the command listings for the description of these commands.

Saving and Loading Settings

Firmware Version **3.71** and above support UPDATED storage of settings to SD card. The setting system has been brought in-line with other Quarch products, so there is no longer a need to explicitly 'save' the settings.

To check if a valid SD card is installed in your Array Controller, use the command “***TST?**”. If a valid card is not present you will be given a warning message. If you have an older Array Controller that did not ship with an SD card, please contact Quarch for a list of compatible cards and fitting instructions.

When you change an internal setting, such as IP address, the setting will persist through reset. To reset back to the factory defaults, use the “**CONFig:DEFault FACTory**” command

Basic Commands

***RST**

Triggers a reset in all connected Array Controllers and Control Modules, the system will behave as if it had just been powered on

***CLR**

Clears the terminal window and displays the normal start screen

***IDN?**

Displays a standard string, identifying the Array Controller in the form:

“Manufacturer, Product part #, Processor part #, Firmware version string”

Where:

Manufacturer = “Quarch Technology”

Product part # = “QTLxxxx” (Part number of the Array Controller)

Processor part # = “QTLxxxx” (Part number of the communicating processor)

Firmware Version = A parsable list containing the version of part

***TST?**

Runs a set of standard tests to confirm the Array Controller is operating correctly, these tests are also performed at start up.

Firmware Version: 3.51

This command now also returns a list of warnings if the Array Controller is missing anything that may be required for operation:

- Checks for serial number and MAC address (required for Ethernet operation)
- Checks for SD card (required for storing persistent settings)

[comments]

Any line beginning with a # character is ignored as a comment. This allows commenting of scripts for use with the system.

General Setup Commands

CONFig: MESSAgEs [SHORT|USER]

CONFig: MESSAgEs?

Gets or sets the mode for messages that are returned to the user's terminal

Short: Only a "FAIL" or "OK" will be returned

User: Full error messages with description are returned to the user on failure

CONFig:TERMiNal [USER|SCRIPT]

CONFig:TERMiNal?

Sets the ASCII terminal configuration mode:

USER: Standard mode for manual control via Hyperterminal etc

SCRIPT: Does not echo characters, sends <CR> after the cursor. This is to assist scripting, especially from Linux based systems

CONFig:TERMiNal:HANDshake [ON|OFF]

CONFig:TERMiNal:HANDshake?

Sets the state of hardware handshaking on the serial terminal. When 'ON', your terminal program must support and use hardware handshaking.

CONFig:TIME?

Returns the current time from an Ethernet SNTP server. If a server is not available, the time starts at "01/01/1970 00:00:00" on power up/reset and increments from there.

CONFig:LIST MODUles?

Lists the active modules in the system. This will only work fully if the attached modules are Firmware 3.50 or above. Otherwise the module response will return an error instead of it's identity.

CONFig:LIST? *(Firmware 3.60 and above)*

Lists ALL items in the system, including Array Controllers. This will only work fully if the attached modules are Firmware 3.50 or above. Otherwise the module response will return an error instead of it's identity.

CONFig:DEFault FACTORY *(Firmware 3.71 and above)*

Resets all stored settings on the Array Controller to its factory default. If an SD card is not present, a warning will be given (the current Array will be reset, but the factory settings cannot be stored)

Addressing Commands

CONFig:MAPping:WRITe [HH] [SS]

Sets the mapping table for hard port [HH] to the soft address [SS]. Both address numbers are simple decimal strings (eg 17). The changes will have no effect until the new mapping is activated.

Example:

Conf:map:write 2 5

Conf:map:write 5 2 (swaps the mapping for ports 2 and 5)

CONFig:MAPping:READ [HH]

Returns the current mapping of the given hard from the mapping table

Example:

Conf:map:read 5

Returns:

5=2 (Hard address 5 is set to soft address 2)

CONFig:MAPping:DUMP [HH] [HH]

Returns the current mapping of the given range of hard addresses, from the mapping table.

Example:

Conf:map:dump 3 5

Returns:

3=3

4=4

5=2

(Hard address 5 is set to soft address 2, others as default)

CONFig:MAPping:ACTivate

Activates the current mapping pattern by setting the soft addresses of all port processors. This must be run in order to use the changes made by the MAPping:WRITe commands.

CONFig:MAPping:RESet

Resets the mapping table to the default mode (as it was at reset) and activates the mapping. This is used to bring the mapping back to a known state. Default state is (Hard Port = Soft Port).

CONFig:MAPping:FLAsh [SS]

Flashes the light of the port. This is used to 'find' the cable that a given address number is plugged into.

Ethernet Commands

CONFig:ETHernet:SNTP [ON|OFF]

CONFig:ETHernet:SNTP?

Changes the state of the SNTP client. Returned states may be “OFF, SEARCHING or LOCKED”. A valid time will not be available from the “Conf:time” command until the status is ‘LOCKED’.

CONFig:ETHernet:IP [xxx.xxx.xxx.xxx] (Firmware 3.60 and above)

CONFig:ETHernet:IP?

Sets the static IP address of the unit. You should then run ‘**config:settings save**’ to save the change to the SD card and reset the unit. This ensures that it appears correctly on the network. Setting a static IP address will disable DHCP.

CONFig:ETHernet:MASK [xxx.xxx.xxx.xxx] (Firmware 3.60 and above)

CONFig:ETHernet:MASK?

Sets the IP Mask of the unit. You should then run ‘**config:settings save**’ to save the change to the SD card and reset the unit. This ensures that it appears correctly on the network.

CONFig:ETHernet NAME?

Returns the NETBIOS name of the device. This is based on the its serial number and is: **QUARCH-T-*nn-*nnn**** where *nn-*nnn** are the last 6 characters of the serial number, found on the back of the unit.

Voltage Measurement Commands

MEASure:VOLTage:SELF [12v?|5v?|3v3?]

Measures the internal voltage rails of the Array Controller. Measurements are returned in the form "12000mV"

User Access Control Commands

These commands are only relevant for ReST control, they will have no effect on USB, Serial or Telnet interfaces.

CONFig:USER:CONtrol [ON|OFF]

CONFig:USER:CONtrol?

Sets or returns the user control enable status. When enabled, ReST commands MUST provide a UserID. When disabled, ReST commands MUST NOT provide a UserID

CONFig:USER:CLEAR

Deletes all existing users. **WARNING:** If you run this command while access control is enabled, you will no longer be able to use ReST control until you turn it off or add new users via another communications method.

CONFig:USER:DUMP [#start] [#end]

Dumps the user data on one or more users, where #start and #end are integer numbers from 0 to 14

The dump format is:

```
USER INDEX: 2  
USER ID : mike1002  
1111 0000 0000  
0000 0000 0000  
1000  
ADMIN
```

This user is in slot position 2, has a UserID of 'mike1002', has access to ports 1-4 and 25 and is also an admin user.

CONFig:USER:SET:[#UserIndex]:SET [#UserID] [ALL]*

Sets up a user in a given slot. #UserIndex is an integer from 0-14. #UserID is an 8 character alphanumeric and 'ALL' is an optional parameter which will assign control to all ports (otherwise the user defaults to no port access).

If you use this command on a UserIndex that already has a user in it, the previous user will be overwritten, removing them.

CONFig:USER:[#UserIndex]:DROP

Clears the user in a given slot. #UserIndex is an integer from 0-14. The selected user will be deleted and the UserID will no longer work

CONFig:USER:[#UserIndex]:GRANT [#Start] [#End]

Grants the user at the given #UserIndex (integer 0-14) location with access to the ports from #Start to #End, inclusive. Ports on the controller are numbered 1 to 28

CONFig:USER:[#UserIndex]:REVOKE [#Start] [#End]

Revokes permissions for the user at the given #UserIndex (integer 0-14) location, blocking access to the ports from #Start to #End, inclusive. Ports on the controller are numbered 1 to 28

CONFig:USER:[#UserIndex]:GRANT ADMIN

Grants the user at the given #UserIndex (integer 0-14) location with Administrator access. This allows use of commands that will affect the entire array (such as reset commands). It also allows the user to run the CONFig:USER command set, granting the ability to change user permissions and add/delete users.

Voltage Measurements

The Array Controller is capable of measuring various voltages for self test. The following measurement points are available:

Measurement Command	Description	Resolution / Accuracy
MEASure:VOLTage:SELF 12v?	Returns the voltage of the Array Controllers internal 12v power rail	64mV / 3%
MEASure:VOLTage:SELF 5v?	Returns the voltage of the Array Controllers internal 5v power rail	16mV / 3%
MEASure:VOLTage:SELF 3v3?	Returns the voltage of the Array Controllers internal 3.3v power rail	12mV / 3%

Factory Default Startup State

On power up or reset, the Array Controller enters a default state. If an SD card is installed, the settings will be loaded from this (Requires firmware v3.60 and higher). The factory default settings are below:

Setting	Default State
Addressing Mode	Hard mapped addressing
Soft Address Map	Ports numbered sequentially from 1 (See "Address Mapping" section above)
Terminal Mode	"User" mode, default for use with HyperTerminal
Message Returns	"User" mode; detailed error responses
IP Address	Obtained through DHCP, or 192.168.1.99

Control Register Map

The Array Controller register map is made available here for debug purposes only, it should never normally be necessary to access them.

Address	Name	Description
0x00	Override Register	Override Mux Settings
0x01	Port 1 Control	Port 1 Status & Enable
0x02	Port 2 Control	Port 2 Status & Enable
0x03	Port 3 Control	Port 3 Status & Enable
0x04	Port 4 Control	Port 4 Status & Enable
0x05	Port 5 Control	Port 5 Status & Enable
0x06	Port 6 Control	Port 6 Status & Enable
0x07	Port 7 Control	Port 7 Status & Enable
0x08	Port 8 Control	Port 8 Status & Enable
0x09	Port 9 Control	Port 9 Status & Enable
0x0A	Port 10 Control	Port 10 Status & Enable
0x0B	Port 11 Control	Port 11 Status & Enable
0x0C	Port 12 Control	Port 12 Status & Enable
0x0D	Port 13 Control	Port 13 Status & Enable
0x0E	Port 14 Control	Port 14 Status & Enable
0x0F	Port 15 Control	Port 15 Status & Enable
0x10	Port 16 Control	Port 16 Status & Enable
0x11	Port 17 Control	Port 17 Status & Enable
0x12	Port 18 Control	Port 18 Status & Enable
0x13	Port 19 Control	Port 19 Status & Enable
0x14	Port 20 Control	Port 20 Status & Enable
0x15	Port 21 Control	Port 21 Status & Enable
0x16	Port 22 Control	Port 22 Status & Enable
0x17	Port 23 Control	Port 23 Status & Enable
0x18	Port 24 Control	Port 24 Status & Enable
0x19	Port 25 Control	Port 25 Status & Enable
0x1A	Port 26 Control	Port 26 Status & Enable
0x1B	Port 27 Control	Port 27 Status & Enable
0x1C	Port 28 Control	Port 28 Status & Enable
0x1D	Port 29 Control	Port 29 Status & Enable
0xFE	Mux2 Version	FPGA firmware version
0xFF	Mux1 Version	FPGA firmware version

Register Definitions

0x00 - Override Register

7	6	5	4..0
OVERRIDE	0	0	OVERRIDE_PORT

OVERRIDE	Read/Write	Bit is set to lock the serial mux onto a specific array port
OVERRIDE_PORT	Read/Write	Port number to lock the mux to in override mode.

0x01..0x1D - Port Control

7	6	5..1	0
RTS	CTS	00000	ENABLE_PORT

RTS	Read Only	Real time state of RTS for this port, a '1' indicates that this port is waiting to retrun data
CTS	Read Only	Real time state of CTS for this port, a '1' indicates that this port is in the process of returning data
ENABLE_PORT	Read/Write	Clear this bit to stop the port from automatically returning data (then it will only be accessible by use of the override register)

0xFE - Mux 2 Version

7..0
VERSION

VERSION	Read	Should read 0x14 (v1.4) or Higher
---------	------	-----------------------------------

0xFF - Mux 1 Version

7..0
VERSION

VERSION

Read

Should read 0x14 (v1.4) or Higher